



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **08212038 A**

(43) Date of publication of application: 20 . 08 . 96

(51) Int. Cl.

G06F 3/14(21) Application number: **07016122**(22) Date of filing: **02 . 02 . 95**(71) Applicant: **PFU LTD**

(72) Inventor: **NISHIDA BARAKO
MURATA KOICHI
SAKAI HIROTAKE
OKUDA KATSUMI**

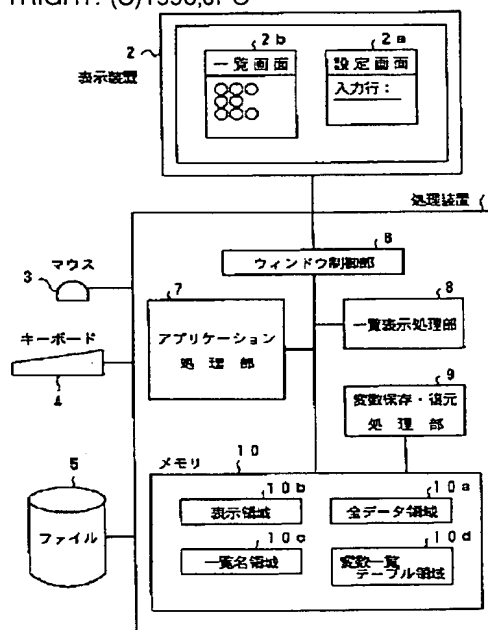
(54) **DISPLAY PROCESSING METHOD FOR LIST PICTURE**

COPYRIGHT: (C)1996,JPO

(57) Abstract:

PURPOSE: To simply and quickly display necessary list data on a screen by reading out a stored list name and displaying its corresponding list picture at the time of executing prescribed control instructing operation for a list specifying area.

CONSTITUTION: An application processing part 7 interactively executes application processing such as table preparation and table calculation. A list display processing part 8 displays a 1st name set up in a list specifying area on a set screen 2a and the list data of a list name stored in a 1st name area 10c and being displayed immediately before current processing on a list screen 2b in accordance with instructing operation from a mouse 3 or a keyboard 4. The list data are written in an all-data area 10a and transferred from the area 10a to the display area to display the data on the screen. The display data on the list screen 2b can be scrolled and the size of a picture, the size of a display part, the font of characters, the display range of data, etc., can be mutually adjusted and changed.



(57) [Summary]

[Object] To quickly display required list data on a screen by a simple operation.

[Configuration] With respect to list data repeatedly used, information on a displayed list window is stored to be used when the window is again displayed, thus enabling display of the same list window by a simple operation. The arrangement is also such that starting/stopping display of a list window by a one-touch key operation is possible. The area for display list data is automatically set restrictively so as to be effective with respect to the kind of window currently displayed and the state of selection of items. Display is effectively performed by changing the sizes of a list window and a display area, the area for list data to be displayed, and fonts in a state of being related to each other. Display of a list of a hierarchical structure is such that at the time of return from a child layer to a parent layer, the list window of the parent layer is restored to the original window state.

[0045]

Fig. 8 is a diagram for explaining window reconfiguration processing with respect to cases where at least one of a window, a component and a font is changed in size. The section (a) of

Fig. 8 shows an example of processing for changing the size of a component and a display data area when the size of a window is changed. In the illustrated example, an operation of increasing the size of a window 18 in each of the vertical and horizontal directions is performed. The size of a component 19 is correspondingly increased and the display area is increased by effectively utilizing an increased display space so that data portions in the right and bottom sides unseen in the preceding state become visible.

[0046]

In the example shown in (a) of Fig. 8, a window mode is set in which the amount of displayed data is not fixed when an operation of changing the window size is performed. In this case, only the size of the component in which list data is displayed is changed and processed with the change in the window size. List data is cut out from the display memory by using the changed component size, thereby automatically forming the window as illustrated.

[0047]

In contrast, the section (b) of Fig. 8 shows processing in a case where a window mode is set in which the amount of displayed data is fixed when an operation of changing the window size is performed. In this case, while the component size is changed with the change in the window size, the

character font size is changed in accordance with a predetermined rule in such a manner that the constant amount of list data to be displayed is placed in the display space of the changed component size so as to be easy to see. In the illustrated example, the font size is automatically changed so that the display layout in the component before the change of the window size is preserved as faithfully as possible.

[0048]

The section (c) of Fig. 8 shows window reconfiguration processing in a case where a change in the font size of displayed characters to be displayed is designated in a window mode in which the amount of data displayed in the window is fixed. In this case, processing for automatically changing the sizes of the component 19 and the window 18 is performed so that the layout of the displayed data in the form of characters of a changed font size is preserved as faithfully as possible.

[0049]

Fig. 9 shows an example in a case different from those shown in (a), (b) and (c) of Fig. 8, i.e., a case where an operation for changing the component size is first performed. An end of the component 19 is dragged with a mouse cursor 20 to change the component size. In this case, a window mode is set in which the amount of displayed data is not fixed. When the component size is determined, the window size is changed

according to the component size to display an amount of data increased to the limit determined by the display area of the component.

[0050]

The sections (a), (b), (c), and (d) of Fig. 10 are flowcharts respectively showing the window reconfiguration processings shown in (a), (b), and (c) of Fig. 8 and Fig. 9. The flowchart in (a) of Fig. 10 shows processing for changing the window size in the window mode in which the amount of displayed data is not fixed. In this processing, changes in the component size according to the changed window size are computed and the changed component is reconfigured in the window.

[0051]

The flowchart in (b) of Fig. 10 shows processing for changing the component size in the window mode in which the amount of displayed data is fixed. In this processing, a size of the component according to the changed window size is computed, a font size according to the computed component size is computed, the changed component is reconfigured in the window, and the changed font is set.

[0052]

The flowchart in (c) of Fig. 10 shows processing for changing the font size in the window mode in which the amount

of displayed data is fixed. A size of the component and a size of the window for setting a display area necessary for displaying the fixed amount of data with a designated font size are computed, the component of the changed size is reconfigured in the window of the changed size according to the results of the computation, and the font of the designated size is set.

[0053]

The flowchart in (d) of Fig. 10 shows processing for changing the component size in the window mode in which the amount of displayed data is not fixed. A size of the window corresponding to a designated component size is computed and the designated component is reconfigured in the window of the size thereby computed.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-212038

(43) 公開日 平成8年(1996)8月20日

(51) Int.Cl.⁶

G 0 6 F 3/14

識別記号

3 4 0 C

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数14 O L (全 19 頁)

(21) 出願番号 特願平7-16122

(22) 出願日 平成7年(1995)2月2日

(71) 出願人 000136136

株式会社ビーエフユー

石川県河北郡宇ノ気町宇野気ヌ98番地の
2

(72) 発明者 西田 パラ子

大阪府大阪市北区堂島1丁目5番17号 ビ
ーエフユー関西ソフトウェア株式会社内

(72) 発明者 村田 浩一

大阪府大阪市北区堂島1丁目5番17号 ビ
ーエフユー関西ソフトウェア株式会社内

(74) 代理人 弁理士 長谷川 文廣 (外2名)

最終頁に続く

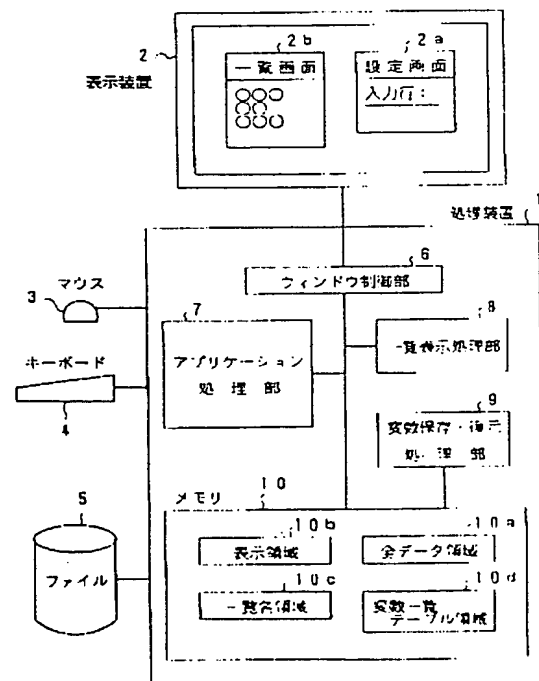
(54) 【発明の名称】 一覧画面の表示処理方法

(57) 【要約】

【目的】 必要とされる一覧データを簡単な操作で迅速に画面表示できるようにすることを目的としている。

【構成】 反復して使用される一覧データについては、表示された一覧画面の情報を記憶しておいて、次に表示する場合にはその情報を利用することにより同一一覧画面を簡単な操作で表示できるようにし、またキー操作によるワンタッチでの一覧画面の表示／表示終了を可能にし、また表示する一覧データの範囲は現在表示されている画面の種類や項目の選択状態に関連して有効性の高いものに自動的に絞り込むようにし、また一覧画面や表示領域のサイズと表示する一覧データの範囲、フォント、などを関連付けて変更して効率的に表示を行い、また階層構造の一覧では、子階層から親階層に戻る場合、親階層の一覧画面を元の画面状態に復元する、等の構成をもつ。

本発明実施例による処理システムの基本構成図



【特許請求の範囲】

【請求項1】 一覧画面の表示処理方法において、表示させたい一覧を指定するための一覧指定領域に一覧名が設定され所定の起動操作が行われたとき、対応する一覧画面を表示し、次に一覧画面が終了されたときその一覧名を記憶しておき、その後上記一覧指定領域に対して所定の制御指示操作が行われたとき、上記記憶されている一覧名を読み出して対応する一覧画面を表示することを特徴とする一覧画面の表示処理方法。

【請求項2】 指定されたデータベースに含まれる全ての表の表名の一覧を表示する表名の一覧画面の表示処理方法において、

表名の一覧画面を呼び出すパラメタを設定するパラメタ設定画面を表示し、パラメタ設定画面にデータベース名が設定され、続いて所定の起動操作が行われたとき、そのデータベース名を記憶するとともに、対応するデータベースの表名の一覧画面を表示し、次に表名の一覧画面が終了された後に表示された上記パラメタ設定画面において所定の制御指示操作が行われたとき、上記記憶されているデータベース名を読み出して対応するデータベースの表名一覧画面を表示することを特徴とする表名の一覧画面の表示処理方法。

【請求項3】 表名の一覧画面の表示処理方法において、

表名の一覧画面の表示を指示するキーおよび表名の一覧画面の終了を指示するキーを設け、これらのキーの操作にตอบสนองして、表名の一覧画面を表示しあるいは表示されている表名の一覧画面を終了させることを特徴とする表名の一覧画面の表示処理方法。

【請求項4】 一覧選択画面で選択された一覧名の一覧画面を表示する一覧画面の表示処理方法において、一覧選択画面に表示する一覧名の範囲を、現在表示されている画面の種類に応じて変更することを特徴とする一覧画面の表示処理方法。

【請求項5】 一覧画面の表示処理方法において、一覧画面のサイズを変更する操作が行われたとき、変更された一覧画面のサイズに応じて該一覧画面内に表示する一覧データの範囲を変更し、画面を再構成することを特徴とする一覧画面の表示処理方法。

【請求項6】 一覧画面の表示処理方法において、一覧画面のサイズを変更する操作が行われたとき、変更された一覧画面のサイズに応じて該一覧画面内に表示する一覧データの文字のフォントサイズを変更し、画面を再構成することを特徴とする一覧画面の表示処理方法。

【請求項7】 一覧画面の表示処理方法において、一覧画面を構成する表示部品のサイズを変更する操作が行われたとき、変更された表示部品のサイズに応じて該一覧画面のサイズを変更し、画面を再構成することを特徴とする一覧画面の表示処理方法。

【請求項8】 階層構造をもつ一覧画面の表示処理方法

において、

親階層の一覧画面で任意の行位置の項目が選択され、該項目に対応する子階層の一覧画面が起動されたとき、現在表示されている親階層の一覧画面の状態を記憶した後子階層の一覧画面を表示し、その後親階層の一覧画面へ復帰する指示操作が行われたとき、先に記憶してある親階層の一覧画面の状態に基づき親階層の一覧画面を表示することを特徴とする一覧画面の表示処理方法。

【請求項9】 一覧の全データ量が表示メモリ領域の大きさを超過しており、一覧の全データが格納されているメモリの全データ領域からメモリの表示領域に、表示対象となる一覧データの部分を必要になる都度複写して一覧画面の表示を行う一覧画面の表示処理方法において、

一覧画面への一覧データの表示要求が、表示領域に格納されている一覧データの最後のデータから後のデータあるいは先頭のデータから前のデータに対してなされたとき、それぞれ、表示に必要となるデータの先頭よりも複数行だけ前のデータが表示領域の先頭データとなるように、あるいは表示に必要となるデータの最後よりも複数行だけ後のデータが表示領域の最終データとなるように、全データ領域から一覧データの複写を行うことを特徴とする一覧画面の表示処理方法。

【請求項10】 請求項9において、一覧画面に表示可能な一覧データの量は、表示メモリ領域に格納されている一覧データの量と同じかそれよりも少なく、一覧画面に表示される一覧データの範囲は、スクロール操作指示に応じて変更されることを特徴とする一覧画面の表示処理方法。

【請求項11】 請求項10において、スクロール操作指示では、スクロールする行数単位を指定し、指定された行数単位でスクロールを行うことを特徴とする一覧画面の表示処理方法。

【請求項12】 請求項1ないし請求項11において、一覧画面上でヘルプ表示操作が行われたとき、一覧画面でカーソルにより選択されている行位置の項目に関するヘルプ情報を表示することを特徴とする一覧画面の表示処理方法。

【請求項13】 変数の一覧画面の表示処理方法において、

変数の一覧画面が表示されている状態でアプリケーション処理が実行され、実行結果により変数の値が決定されたとき、該変数の値を上記変数の一覧画面に直ちに反映させ、変数の一覧画面を更新することを特徴とする変数の一覧画面の表示処理方法。

【請求項14】 変数の一覧画面の表示処理方法において、

変数の一覧画面に表示される変数の値の一覧データを登録した変数一覧テーブルの内容を、処理の中断あるいは所定の保存指示操作にตอบสนองしてファイルに格納して保存し、また所定の復元指示操作にตอบสนองして上記ファイルに

格納されている内容を取り出して変数一覧テーブルを元の状態に復元することと特徴とする変数の一覧画面の表示処理方法。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、画面との対話により表データの入力、表計算などを実行する処理システムにおいて、利用できる各種の一覧を迅速容易に表示するための一覧画面の表示処理方法に関する。

【0002】多数のデータや文書を管理し、迅速に参照可能にする手段の1つとして一覧がしばしば利用される。この一覧には、処理上使用できる変数値などを書き込んだメモ一覧や、バインダメモ一覧、ユーザメモ一覧、DB名（データベース名）一覧、表名一覧、DBメモ一覧などがあり、アプリケーションの処理中、必要に応じて呼び出し、ウィンドウ画面に表示される。

【0003】しかし一覧の種類が多くなり、また1つの一覧に含まれるデータの量が多くなると、参照したい一覧を指定してその一覧画面を呼び出し、さらに一覧中の見たいデータ部分を画面中表示させるのに要する操作負担が大きくなり、時間もかかるようになる。本発明は、このような一覧画面に対する効率的な表示処理方法を提供する。

【0004】

【従来の技術】従来の処理システムでは、データの一覧を表示させる場合、以下のような問題があった。

【0005】(1) 表示したいデータの一覧を指定するには、図23(a)に示すように、画面上に設けられている一覧指定領域43に表示したい一覧の名前を設定するか、図23(b)に示すように、一覧の名前を並べた一覧選択画面44を表示して、その中からカーソルで目的の一覧を選択する操作を行う必要があり、同じ一覧画面を何度も表示させるような場合、そのたびに同じ一覧名を一つ指定あるいは選択しなければならず、手間が掛かっていた。

【0006】(2) DB（データベース）内の表を知りたい場合、DB名を指定して表名一覧という画面を表示させ、その中から目的の表名を選択する必要があった。この表名一覧画面は何らかの操作により閉じてしまった時に、再度表示させるにはまたDB名を指定しなければならず、面倒であった。

【0007】(3) あるOS(Microsoft Windowsなど)では、一覧画面をウィンドウに表示するには、〔ファイル〕－〔開く〕等のメニューを選択してウィンドウを開き一覧画面を表示し、〔ファイル〕－〔閉じる〕等のメニューを選択してウィンドウを閉じるのが標準的な操作方法となっており、表名一覧画面のように一時的に使用されるウィンドウでは大変わずらわしい操作となっている。

【0008】(4) 一覧の種類が多くなるとユーザがそれ

らを覚えるのは困難になり、一覧選択画面を表示しても多数の一覧名の中から必要とする一覧名を探すのはむずかしくなる。

【0009】(5) 一覧画面はきまった大きさの画面サイズで表示される場合が多く、表示させたいデータの量に対して表示領域が有効に利用されない場合があった。

(6) 階層構造をもつ一覧では、図24に示すように、親階層の一覧画面から子階層の一覧画面に移った後、親階層の一覧画面に戻ると、元の画面の状態とは無関係に一覧データが先頭から表示されるので、必要なデータのところまでスクロール操作しなければならないことがあった。

【0010】(7) 大量のデータをもつ一覧の場合、画面の表示領域が小さくてデータが入りきれないと図25に示すように分割して表示されるが、画面の上端、下端近傍でのスクロール操作が増え、その際、表示メモリ領域へのデータ転送頻度が著しく増大することがあった。

【0011】(8) 一覧中の任意の項目についてヘルプ情報を見たい場合、ヘルプ情報の数が多いと目的のヘルプ情報を見付けにくかった。またその場合、検索画面で項目名を入力する必要があるため、面倒であった。

【0012】(9) アプリケーションの処理において、複数の変数に対する計算結果を確認したい場合には、計算結果の各変数の値が登録される変数一覧（メモ一覧の一種）を参照する必要がある。しかし図26に示すように、変数一覧画面が表示されている状態でアプリケーション処理が行われて演算が終了した場合には、その計算結果は変数一覧の表示内容には直ちに反映されず、一旦画面を閉じて再表示の操作を行い、表示メモリ領域の内容を更新させる必要があった。

【0013】(10) 処理の中断時などのアプリケーション処理結果の各変数の値を保存する必要がある場合、従来は変数ごとにファイルに書き込む操作が必要であり、手間がかかっていた。

【0014】

【発明が解決しようとする課題】本発明は、必要とされる一覧データを簡単な操作で迅速に画面表示できるようにすることを目的としている。

【0015】

【課題を解決するための手段】本発明は、上記課題を解決するために、反復して使用される一覧データについては、表示された一覧画面の情報を記憶しておいて、次に表示する場合にはその情報を利用することにより同じ一覧画面を簡単な操作で表示できるようにし、またキー操作によるワンタッチでの一覧画面の表示、表示終了を可能にし、また表示する一覧データの範囲は現在表示されている画面の種類や項目の選択状態に関連して有効性の高いものに自動的に絞り込むようにし、また一覧画面や表示領域のサイズと表示する一覧データの範囲、フォント、などを関連付けて変更して効率的に表示を行い、ま

た階層構造の一覧では、子階層から親階層に戻る場合、親階層の一覧画面を元の画面状態に復元し、また一覧データの全データ量が表示領域の大きさを超えている場合に、メモリの表示領域の下端、上端のデータを越える表示要求に対しては、それぞれ、下端のデータが表示領域の先頭よりも複数行だけ後の行に、あるいは上端のデータが表示領域の最後よりも複数行だけ前の行に位置するようにデータを入れ替え、スクロールによるデータの入れ替えがひんばんに発生しないようにし、またスクロールする単位を一覧の参照の仕方に応じて適切に設定できるようにして、スクロール操作回数の削減を図り、また変数一覧では、常に最新の状態が表示されるようにするとともに保存、復元を簡単に行えるようにする。

【0016】これに基づく本発明の具体的構成は、以下の態様をとるものである。

(構成1) 一覧画面の表示処理方法において、表示させたい一覧を指定するための一覧指定領域に一覧名が設定され所定の起動操作が行われたとき、対応する一覧画面を表示し、次に一覧画面が終了されたときその一覧名を記憶しておき、その後上記一覧指定領域に対して所定の制御指示操作が行われたとき、上記記憶されている一覧名を読み出して対応する一覧画面を表示することを特徴とする。

【0017】(構成2) 指定されたデータベースに含まれる全ての表の表名の一覧を表示する表名の一覧画面の表示処理方法において、表名の一覧画面を呼び出すパラメタを設定するパラメタ設定画面を表示し、パラメタ設定画面にデータベース名が設定され、続いて所定の起動操作が行われたとき、そのデータベース名を記憶するとともに、対応するデータベースの表名の一覧画面を表示し、次に表名の一覧画面が終了された後に表示された上記パラメタ設定画面において所定の制御指示操作が行われたとき、上記記憶されているデータベース名を読み出して対応するデータベースの表名一覧画面を表示することを特徴とする。

【0018】(構成3) 表名の一覧画面の表示処理方法において、表名の一覧画面の表示を指示するキーおよび表名の一覧画面の終了を指示するキーを設け、これらのキーの操作に応答して、表名の一覧画面を表示しあるいは表示されている表名の一覧画面を終了させることを特徴とする。

【0019】(構成4) 一覧選択画面で選択された一覧名の一覧画面を表示する一覧画面の表示処理方法において、一覧選択画面に表示する一覧名の範囲を、現在表示されている画面の種類に応じて変更することを特徴とする。

【0020】(構成5) 一覧画面の表示処理方法において、一覧画面のサイズを変更する操作が行われたとき、変更された一覧画面のサイズに応じて該一覧画面内に表示する一覧データの範囲を変更し、画面を再構成す

ることを特徴とする。

【0021】(構成6) 一覧画面の表示処理方法において、一覧画面のサイズを変更する操作が行われたとき、変更された一覧画面のサイズに応じて該一覧画面内に表示する一覧データの文字のフォントサイズを変更し、画面を再構成することを特徴とする。

【0022】(構成7) 一覧画面の表示処理方法において、一覧画面を構成する表示部品のサイズを変更する操作が行われたとき、変更された表示部品のサイズに応じて該一覧画面のサイズを変更し、画面を再構成することを特徴とする。

【0023】(構成8) 階層構造をもつ一覧画面の表示処理方法において、親階層の一覧画面で任意の行位置の項目が選択され、該項目に対応する子階層の一覧画面が起動されたとき、現在表示されている親階層の一覧画面の状態を記憶した後に子階層の一覧画面を表示し、その後親階層の一覧画面へ復帰する指示操作が行われたとき、先に記憶してある親階層の一覧画面の状態に基づき親階層の一覧画面を表示することを特徴とする。

【0024】(構成9) 一覧の全データ量がメモリの表示領域の大きさを超えており、一覧の全データが格納されているメモリの全データ領域から表示領域に、表示対象となる一覧データの部分を必要になる都度複写して一覧画面の表示を行う一覧画面の表示処理方法において、一覧画面への一覧データの表示要求が、表示メモリ領域に格納されている一覧データの最後のデータから後のデータあるいは先頭のデータから前のデータに対してなされたとき、それぞれ、表示に必要となるデータの先頭よりも複数行だけ前のデータが表示領域の先頭データとなるように、あるいは表示に必要となるデータの最後よりも複数行だけ後の、全データ領域から一覧データの複写を行うことを特徴とする。

【0025】(構成10) 構成9において、一覧画面に表示可能な一覧データの量は、表示メモリ領域に格納されている一覧データの量と同じかそれよりも少なく、一覧画面に表示される一覧データの範囲はスクロール操作指示に応じて変更されることを特徴とする。

【0026】(構成11) 構成10において、スクロール操作指示では、スクロールする行数単位を指定し、指定された行数単位でスクロールを行うことを特徴とする。

【0027】(構成12) 構成1ないし構成11において、一覧画面上でヘルプ表示操作が行われたとき、一覧画面でカーソルにより選択されている行位置の項目に関するヘルプ情報を表示することを特徴とする。

【0028】(構成13) 変数の一覧画面の表示処理方法において、変数の一覧画面が表示されている状態でアプリケーション処理が実行され、実行結果により変数の値が決定されたとき、該変数の値を上記変数の一覧画面に直ちに反映させ、変数の一覧画面を更新することを

特徴とする。

【0029】（構成14） 変数の一覧画面の表示処理方法において、変数の一覧画面に表示される変数の値の一覧データを登録した変数一覧テーブルの内容を、処理の中断あるいは所定の保存指示操作に応答してファイルに格納して保存し、また所定の復元指示操作に応答して上記ファイルに格納されている内容を取り出して変数一覧テーブルを元の状態に復元することを特徴とする。

【0030】

【作用】本発明の各方法を適用することにより、以下の10
ような作用が生じる。

a. 一覧画面を表示させるのに必要な操作量と時間が削減される。

【0031】 b. 一覧画面のサイズに応じて表示領域を有効に利用したデータ表示が行われる。

c. 有効性の高い一覧データが優先して表示され、無駄なデータの表示が抑制される。

【0032】 d. 一覧データのデータ量にくらべて一覧画面に表示できるデータ量が少ない場合でも、表示メモリへのデータ転送回数を少なくできる。

e. 変数一覧画面のデータの最新性が保証される。

【0033】 f. 変数一覧のデータを必要時に退避することにより、処理中断後の再開などが容易になる。

【0034】

【実施例】図1は、本発明実施例による処理システムの基本構成図である。図において、1は処理装置、2は表示装置、2aは設定画面、2bは一覧画面、3はマウス、4はキーボード、5はファイル、6はウィンドウ制御部、7はアプリケーション処理部、8は一覧表示処理部、9は変数保存・復元処理部、10はメモリ、10aは全一覧データが格納される全データ領域、10bは画面表示データが格納される表示領域、10cは一覧名領域、10dは変数一覧テーブル領域である。

【0035】ウィンドウ制御部6はアプリケーション処理部7あるいは一覧表示処理部8からの要求に応じてウィンドウを開きあるいはウィンドウを閉じる。またマウス3あるいはキーボード4からの制御指示に応じてウィンドウのサイズや位置を変更する。

【0036】アプリケーション処理部7は、処理画面により対話形式で表作成、表計算などのアプリケーション処理を行う。一覧表示処理部8は、マウス3あるいはキーボード4からの指示操作に応じて、設定画面2aの一覧名指定領域に設定された一覧名や一覧名領域10cに保持されている直前に表示されていた一覧名の一覧データを一覧画面2bに表示する。一覧データは全データ領域10aに書き込まれており、ここから表示領域10bに転送して画面表示される。一覧画面2bの表示データはスクロールされることができ、また画面サイズや表示部品のサイズ、文字のフォント、データの表示範囲などは相互に調和させて変更することができる。

【0037】変数保存・復元処理部9は、アプリケーション処理部7における処理で設定されたあるいは計算結果の変数値を変数一覧テーブル領域10dで管理し、保存指示操作に応じてテーブルをファイル5に退避させて保存し、また復元指示操作に応じてファイル5から変数一覧テーブル領域10dに復元する。変数一覧テーブルの内容は、一覧表示処理部8によって変数一覧画面として表示されることができる。

【0038】図2は、本発明実施例による一覧画面の表示処理例を示す。図2において、11は表示画面、12は表示したい一覧を指定するための一覧指定領域、13は表示された一覧画面、14は一覧指定領域12に設定された一覧名を記憶する一覧名記憶部、15はカーソルである。また(a)～(d)は画面状態の変化を示す。

【0039】(a)は、キー入力するなどして一覧指定領域12に設定された一覧名により一覧画面13が表示されている画面状態である。ここで画面終了が指示されると一覧画面13が消され、(b)の画面状態となる。このとき、一覧指定領域12に設定されていた一覧名は一覧名記憶部14に設定される。この後(c)のように、一覧指定領域12をカーソル15で選択し、マウスのダブルクリックあるいは特定のキー操作を行うなどの所定の制御指示を与えると、一覧名記憶部14に保持されていた一覧名を用いて一覧画面を表示する処理が行われ、(d)の画面状態となる。

【0040】図3は、図2の実施例に対応する処理のフローである。図3において、まず図23の(b)に示されるような一覧選択画面を表示して一覧名を選択するかまたは表示したい一覧をキー入力する。ここで一覧画面の起動が行われると一覧名に対応する一覧画面13を表示し、また一覧名は図2の一覧名記憶部14に記憶される。一方、カーソルで一覧指定領域12が選択され、ダブルクリックされると、一覧名記憶部14に記憶されている一覧名を用いて一覧画面が表示される。

【0041】図4は、図2において表名一覧画面を表示させる場合の具体例を示す。表名一覧が表示されたときそのDB名（データベース名）をDB名記憶部に記憶しておき、表名一覧画面が閉じられた後、ダイアログ（設定画面）の入力フィールドをダブルクリックするか、

【F11】キーを押下すると、DB名記憶部に記憶されているDB名を用いて直前に表示された表名一覧画面が表示される。表名一覧画面を閉じる場合は〔ESC〕キーを押下する。〔F11〕キーと〔ESC〕キーを用いることにより、一覧画面をワンタッチで表示したり閉じたりすることができる。

【0042】図5は、これらのキーを用いて一覧画面の表示を制御する場合のフローである。キーまたはマウスからの入力があると、〔F11〕キーまたは〔ESC〕キーの入力でなければ、入力されたキー、マウスの対応する処理を行う。しかし〔F11〕キーが押下された場

合は、DB名記憶部に保持されているDB名により、直前に表示されていた表名一覧を表示する。また〔ESC〕キーが押下された場合は、現在表示中の表名一覧画面を閉じ、その表名一覧のDB名をDB名記憶部に設定する。

〔0043〕図6は、本発明実施例による一覧選択画面を表示する処理の説明図である。一覧選択画面の限られたスペースに多種類の一覧を効率的に表示するため、現在ユーザがアクティブにしているウインドウ画面にある表の表種別との関連により、できるだけ有効性の高いものを表示する。そのため図中の一覧管理テーブル16は、一覧ごとに各表種別に共通に表示するものと、特定の表種別についてだけ表示あるいは非表示とするものに分けて管理する。図示の例では、汎用性の高いメモ、バインダメモ、ユーザメモ、DBメモ、関数、DB名の各一覧を共通表示とし、項目名および処理文の一覧を選択表示としている。また項目名一覧は、データ表画面と自動定義表画面のそれぞれがアクティブ状態のとき表示されるが、処理文一覧は自動定義表画面で主に参照され、データ表画面のアクティブ状態では不要のため非表示となる。この結果、データ表画面のアクティブ状態では処理文一覧なしの一覧選択画面17が表示され、自動定義表画面のアクティブ状態では処理文一覧を含む一覧選択画面17'が表示される。

〔0044〕図7は、図6の一覧選択画面表示処理のフローである。一覧選択画面の表示を指示されると、図6の一覧管理テーブル16を参照し、まず共通表示とされる一覧名のグループを一覧選択画面に設定し、次に選択表示とされる項目名一覧と処理文一覧について、それぞれ現在アクティブ状態にある画面の表種別に対する表示の可否を一覧管理テーブル16でチェックし、表示可能であれば一覧選択画面に設定する。

〔0045〕図8は、本発明実施例による画面サイズ、部品サイズ、フォント変更による画面再構成処理の説明図である。図8の(a)は画面サイズの変更時に表示される部品のサイズと表示データ範囲を変更する処理例を示す。図示の例では、画面18のサイズが縦方向と横方向に拡大操作される。これに応じて部品19のサイズも拡大され、大きくなった表示スペースを有効に利用して、それまで見えなかった右方と下方のデータ部分も見えるように表示範囲を拡張している。

〔0046〕図8の(a)では、画面サイズが変更操作されたとき表示データ量が固定されない画面モードに設定されている。この場合には画面サイズとともに一覧データが表示される部品のサイズだけが追従して変更処理され、変更された部品サイズを用いて表示メモリから一覧データを切り出し、図示のような画面が自動的に作成される。

〔0047〕これに対して図8の(b)は、画面サイズの変更操作時に表示データ量が固定である画面モードが設

定されている場合の処理を示す。この場合には、変更された画面サイズに追従して部品サイズが変更されるが、変更された部品サイズの表示スペースに、表示する一定量の一覧データが見やすく配置されるように、予め定められた規則にしたがって文字のフォントサイズが変更される。図示の例は、画面サイズの変更前の部品内の表示レイアウトができるだけ保存されるようにフォントサイズを自動的に変更した例である。

〔0048〕図8の(c)は、画面に表示データ量が固定である画面モードにおいて表示される文字のフォントサイズの変更指定が行われた場合の画面再構成処理を示す。この場合は、変更されたフォントサイズの文字で表示データのレイアウトができるだけ保存されるように、部品19と画面18のサイズを自動的に変更する処理を行う。

〔0049〕図9は、図8の(a)、(b)、(c)の場合とは異なり、部品サイズが先に変更操作された場合の例である。部品19の端をマウスカーソル20でドラッグすることにより部品サイズを変更する。この場合表示データ量が固定されていない画面モードにあり、部品サイズが確定すると画面サイズが追従変更され、部品内の表示領域に入るだけのデータを表示する処理が行われる。

〔0050〕図10の(a)、(b)、(c)、(d)は、それぞれ図8の(a)、(b)、(c)および図9の各画面再構成処理のフローを示す。図10の(a)のフローは、画面サイズの変更および表示データ量非固定の画面モードにおける処理であり、変更された画面サイズに応じて変更する部品サイズを算出し、変更した部品を画面に再配置する。

〔0051〕図10の(b)のフローは、部品サイズの変更および表示データ量固定の画面モードにおける処理のフローであり、変更された画面サイズに対応する部品のサイズを算出し、その結果の部品サイズに対応するフォントサイズを算出し、変更した部品を画面に再配置するとともに、変更したフォントを設定する。

〔0052〕図10の(c)のフローは、フォントサイズの変更および表示データ量固定の画面モードにおける処理のフローであり、指定されたフォントサイズで固定量のデータを表示するのに必要な表示領域をもつ部品のサイズおよび画面のサイズを算出し、その結果の変更されたサイズの画面に変更されたサイズの部品を再配置し、次に指定されたサイズのフォントを設定する。

〔0053〕図10の(d)のフローは、部品サイズの変更および表示データ量非固定の画面モードにおける処理のフローである。指定された部品サイズに対応する画面のサイズを算出し、その結果のサイズの画面に指定された部品を再配置する。

〔0054〕このようにして再構成された画面、部品、フォントに基づいて一覧データが表示される。図11は、本発明実施例による階層構造をもつ一覧データの表示処理の説明図である。図において、文書一覧の全デー

タ21に示される文書データはそれぞれ頁データの一覧に展開される階層構造をもつ。全データ21のデータ量が親階層の一覧画面22のサイズに比べて多い場合、一覧の全データ21から部分的に一定量のデータを切り出して親階層の文書一覧画面22に表示する。図示の例では、「文書4、文書5、文書6」が表示されている。またこの画面の表示範囲から外れているデータはスクロール操作に応じて全データ21から切り出すデータの範囲を上下にずらすことにより表示される。親階層の一覧画面22上で任意の文書名の行を選択し、“子階層を表示する”ボタン23を操作する。このとき、親階層の一覧画面22の表示状態を全データ21における表示位置で記憶し、子階層の一覧データ24を表示する。ここで再び親階層の一覧画面22に戻る場合には、“親階層に戻る”ボタン25を操作する。これにより先に記憶してある親階層の一覧画面22の表示状態に基づいて親階層の一覧画面22'を表示する。

【0055】親階層の一覧画面22の表示状態は、全データ21上でのカーソル位置（図示の例では「文書4」）で管理される。このカーソル位置は、図中の親階層カーソル記憶領域26に記憶されるが、親階層の一覧画面22から子階層の一覧画面24に移行する際にカーソル位置を+1して記憶することにより、図示の親階層の一覧画面22'に見られるように、復帰時に「文書4」からではなく次の行のデータ「文書5」から表示して、継続性のある処理を円滑化できる。

【0056】図12は、このような階層構造をもつ一覧データの表示処理のフローである。まず親階層カーソル記憶領域26を初期化する。次に親階層の一覧画面表示し、そのカーソル位置を親階層カーソル記憶領域26に設定する。次に“子階層を表示する”ボタン23を画面に表示してキー、マウスからの入力を待ち、“子階層を表示する”ボタン23が操作されたとき、“親階層に戻る”ボタン25を画面に表示し、次に親階層カーソル記憶領域26のカーソル位置値を+1更新する。次に子階層の一覧を画面に表示し、子階層のカーソルを先頭位置に設定する。ここでキー、マウスからの入力を待ち、“親階層に戻る”ボタン25が操作されたとき、親階層カーソル記憶領域26のカーソル位置に基づいて親階層の一覧を表示し、以後同様な動作を繰り返す。

【0057】図13は、本発明実施例による大量一覧データの表示処理の説明図である。図13の(a)のように、全一覧データA A A ~ Z Z Zが格納されている全データ領域27から、表示のためのデータが8行分格納される表示領域28に先頭のデータA A Aから後の表の一部A A A ~ H H Hを転送し、一覧画面の表示が行われている状態で、スクロールにより表示領域28の終端データH H Hよりも後の部分に対する表示要求が出された場合、表示領域28のたとえば真ん中のデータE E Eから後の8行分のデータE E E ~ L L Lを、図13の(b)に

示すように全データ領域27から表示領域28に転送し、表示する。これによりデータH H Hの前後に複数行のゆとりをもって一覧データを表示でき、データH H Hの近傍を前後に参照するような場合には全データ領域27からのデータ転送回数を減らすことができる。

【0058】図14は、図13の表示処理に基づく一覧画面のスクロール時の動作説明図である。図13と同様に、全データ領域27から表示領域28にデータA A A ~ H H Hが転送された後、4行分のデータ表示量をもつ一覧画面29に表示される。はじめはデータA A A ~ D D Dが表示され、下方へのスクロール操作の結果、図示のように表示領域28内の後半部のデータE E E ~ H H Hが表示されている状態になっている。ここでさらに下方への画面単位のスクロール操作が行われると、表示領域28にはH H Hまでのデータしかないため、データを入れ替える必要がある。このとき表示領域28には、次に表示に必要なデータI I I ~ L L Lの先頭データI I Iよりも複数行だけ前のデータ（この例ではF F F）から後のデータF F F ~ M M Mを全データ領域27から表示領域28へ転送する。そして表示領域28からスクロールの結果としてのデータI I I ~ L L Lを読み出して、一覧画面29に表示する。この場合、表示領域28にはI I Iよりも前のデータF F F ~ H H Hが残っているため、次に上方への行単位のスクロール操作が行われても表示を戻す必要が生じても、表示領域28のデータを入れ替える必要はない。

【0059】同様にして、表示領域28にデータF F F ~ M M Mが保持されており、一覧画面29にはデータF F F ~ I I Iが表示されている状態で上方への画面単位のスクロール操作が行われた場合には、表示領域28にある先頭の複数行のデータF F F ~ H H Hを重複させて入れ替えることにより、その後続いて下方への行単位のスクロール操作が反復されても、表示領域28へのデータ転送は不要となる。

【0060】スクロールは、通常1行単位あるいは1画面単位で行われるが、多量の一覧データ中の遠くの目標データ位置まで迅速に到達したい場合には、あまり効率的ではない。図15はスクロール単位を可変に指定できる実施例を示し、一覧画面29上にスクロール単位の行数を指定する領域を設け、ユーザが状況に応じてスクロール単位を変更できるようにしている。図示の例ではスクロール単位が200行に設定されており、データ000001 ~ 000004が表示されている。一覧画面29で下方へのスクロール操作が行われた結果、一覧画面29にはデータ000200 ~ 000203が表示される。この場合も、表示領域28のデータを入れ替える場合には、実際に表示されるデータの先頭の複数行前にあるデータから一定量のデータを、全データ領域27から表示領域28に転送する。

【0061】図16は、図13の表示処理に対応するフ

ローであり、図14に示される処理の場合も、基本的には適用可能なものである。図のフローでは、一覧画面が表示されてから、キー、マウス等での入力操作により行方向でのカーソル移動が行われる。カーソルが表示領域にある一覧データの最後に位置している場合、表示領域にあるデータが全データの最後のデータでない限り一覧の表示領域の後半部分を前半部分に複写し、代わりに後半部分には全データ領域からその前半部分に連続する一覧データを複写して、次のカーソル移動操作を待つ。一方、カーソルが表示領域にある一覧データの先頭に位置している場合には、表示領域のデータが全データの先頭のデータでない限り、表示領域の前半部分を後半部分にコピーし、代わりに前半部分には、全データ領域からその前半部分に連続する一覧データを複写して、次のカーソル移動操作を待つ。以上の動作が繰り返される。

【0062】図17は、本発明実施例による一覧画面からヘルプ情報を選択表示する処理の説明図である。図17は、関数一覧の例によるヘルプ情報の表示処理機構を示す。この表示処理では、関数一覧画面30において選択された関数の項目に関連するヘルプ情報だけをヘルプ画面31に表示する。このため、関数一覧画面30においてヘルプキー30aが操作されたとき、その時点でカーソル選択されている項目の行位置により、関数一覧データを管理する関数一覧テーブル32を参照してその項目のインデックスを求め、そのインデックスをヘルプID変換テーブル33でヘルプIDに変換する。ヘルプID変換テーブル33は、関数一覧画面上の各項目に対して関連するヘルプ情報のIDを与えるもので、ヘルプIDは、ヘルプ情報テーブル34に登録されているヘルプ情報を管理する情報である。このヘルプIDを用いてヘルプ情報テーブル34を検索し、必要なヘルプ情報を取り出して、ヘルプ画面31に表示する。

【0063】図18は、図17のヘルプ表示処理のフローである。このフローは、キー入力を待ち、ヘルプキーが入力されたとき、関数一覧テーブル32で選択された項目のインデックスを取得し、さらにヘルプID変換テーブル33でヘルプIDに変換し、ヘルプ情報テーブル34から目的のヘルプ情報を取り出して表示する動作を示している。

【0064】図19は、本発明実施例による変数一覧画面の更新表示処理の説明図である。この実施例では、アプリケーション処理で変数計算が行われて計算結果が得られるなど変数データに変化が生じると、表示中の変数一覧画面35内の該当する変数の値を直ちに書き替える。これにより変数一覧画面は、常に最新の情報を表示する。

【0065】変数一覧画面35は、メモリの表示領域36に書き込まれたデータに基づいてリフレッシュ表示される。また表示領域36のデータは、メモリの全データ領域37のデータを書き込んだものである。そのため変

数一覧画面35を更新するには、まず全データ領域37のデータを書き替え、さらにその書き替えたデータに対応するものが表示領域36にもあるかどうかを調べ、あれば表示領域36のデータの書き替えを行う。一覧表示処理部38はこのための処理を行い、全データ領域37のデータが更新されたとき、表示領域36に更新状態を反映させる処理を行う。図示の例では、変数一覧画面35に表示された変数M3の値が100である状態のとき、アプリケーション処理で変数M3が新たに計算され、M3=50が得られると、上述した処理を行って変数一覧画面35'に見られるように変数M3の値を50に更新する。

【0066】図20は、変数一覧画面の更新表示例を示し、図20の(a)は既存の変数データM3の値を変更した例、図20の(b)は新変数DEのデータを追加した例、図20の(c)は、既存の変数XYのデータを削除した例をそれぞれ示している。

【0067】図21は、図19の変数一覧画面の更新表示処理を起動する演算処理のフローである。図21の(a)はユーザが投入した計算コマンドによる演算処理の場合を示し、計算コマンドに伴う計算式を入力した後、演算共通関数を呼び出す。演算共通関数の処理は図21の(c)に示され、入力された計算式を解析し、その演算を実行する。演算実行により変数の値が変更された場合には図19の一覧表示処理部38に通知し、変数一覧画面の表示内容を更新させる。図21の(b)は、図21の(a)の場合とは異なり、処理画面の実行時に、予め画面中の入力行に定義されている処理手続きが演算を指示するものであるときの計算モード設定による演算処理を示し、入力された計算式について図21の(c)に示す演算共通関数を呼び出し、演算結果が出たならばそれを画面の入力行に設定する。

【0068】図22は、本発明実施例による変数一覧の変数の保存、復元処理の説明図である。図22の(a)は、変数一覧の変数をファイルに保存し、またファイルから復元する処理の説明図である。アプリケーションの処理画面39で実行された結果の変数は、逐次、変数一覧テーブル40に登録され、指示により変数一覧画面41に表示することができる。一方、アプリケーションの処理を一時中断するような場合には、保存の指示により変数一覧テーブル40の内容をファイル42に格納し保存する。処理再開時には、指示によりファイル42の内容を読み出して変数一覧テーブル40を復元し、さらに処理画面39の中断時の状態を復元する。これにより直ちに処理を継続させることができる。

【0069】図22の(b)は、保存処理のフローであり、変数一覧テーブルの各変数値をファイル形式に変換しファイルに格納する。図22の(c)は復元処理のフローであり、ファイルからデータを読み出し、ファイル形式を内部形式に変換して変数一覧テーブルを復元し、さ

らに処理画面に各変数値を反映させる。

【0070】

【発明の効果】本発明によれば、各種一覧の表示処理において、直前の表示情報の記憶の利用や現在の表示画面の情報の利用により、簡単な操作で必要な情報の一覧を迅速に表示させることができ、一覧画面のサイズやフォントサイズ、データ量の一つを指定して他の要素を自動的に変更することができるため、画面内のデータの見やすさや、マルチウインドウでの他の画面との重なりなどを考慮した適切な表示が可能であり、また一覧データのデータ量が多い場合や階層構造をもつ場合のデータ参照を効率化でき、さらに変数に変更、追加、削除された場合に変数一覧画面がリアルタイムで更新されるため最新の処理状態での変数を利用して処理を進めていくことができ、また処理中断時の変数値を保存、復元できるため処理の中断、再開を容易に行うことができる。

【図面の簡単な説明】

【図1】本発明実施例による処理システムの基本構成図である。

【図2】本発明実施例による一覧画面の表示処理例の説明図である。

【図3】一覧画面の表示処理のフローである。

【図4】表名一覧画面表示の具体例の説明図である。

【図5】専用キーによる一覧画面の表示制御のフローである。

【図6】本発明実施例による一覧選択画面の表示処理の説明図である。

【図7】一覧選択画面表示処理のフローである。

【図8】本発明実施例による画面サイズ、部品サイズ、フォントの変更を行う画面再構成処理の説明図である。

【図9】部品サイズの変更指定による画面再構成例の説明図である。

【図10】画面再構成処理のフローである。

【図11】本発明実施例による階層構造をもつ一覧データの表示処理の説明図である。

【図12】階層構造をもつ一覧データの表示処理のフローである。

【図13】本発明実施例による大量一覧データの表示処理の説明図である。

【図14】大量一覧データの表示画面のスクロール時の動作説明図である。

【図15】スクロール単位を任意に指定できる実施例の説明図である。

【図16】大量一覧データの表示処理のフローである。

【図17】ヘルプ情報選択表示処理の説明図である。

【図18】ヘルプ情報選択表示処理フローである。

【図19】本発明実施例による変数一覧画面の更新表示処理の説明図である。

【図20】変数画面の表示例の説明図である。

【図21】変数一覧画面の更新表示処理を起動する演算処理のフローである。

【図22】本発明実施例による変数一覧の変数の保存、復元処理の説明図である。

【図23】従来のデータ一覧画面の表示指定方法の説明図である。

【図24】従来の階層構造をもつ一覧画面の表示操作例の説明図である。

【図25】従来の大量一覧データの表示処理の説明図である。

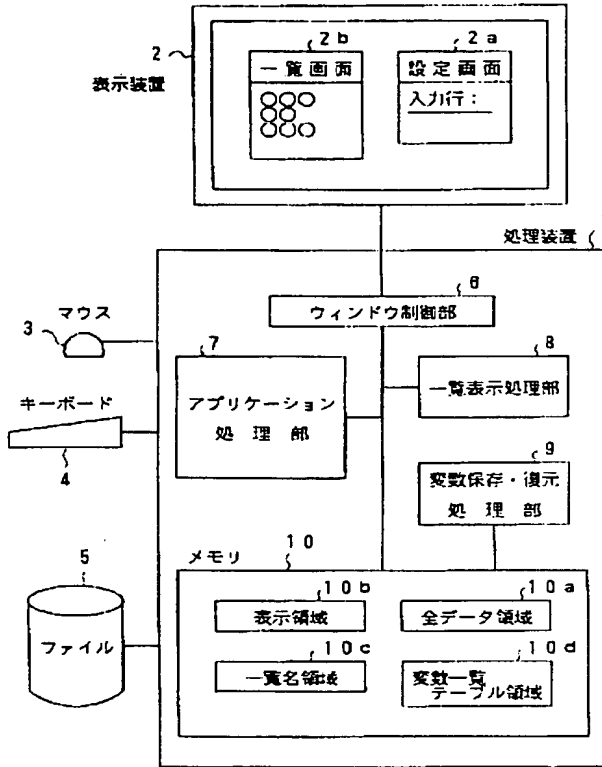
【図26】従来の変数一覧画面の表示処理の説明図である。

【符号の説明】

- 1 処理装置
- 2 表示装置
- 2 a 設定画面
- 2 b 一覧画面
- 3 マウス
- 4 キーボード
- 5 ファイル
- 6 ウインドウ制御部
- 7 アプリケーション処理部
- 8 一覧表示処理部
- 9 変数保存・復元処理部
- 10 メモリ
- 10 a 全データ領域
- 10 b 表示領域
- 10 c 一覧名領域
- 10 d 変数一覧テーブル領域

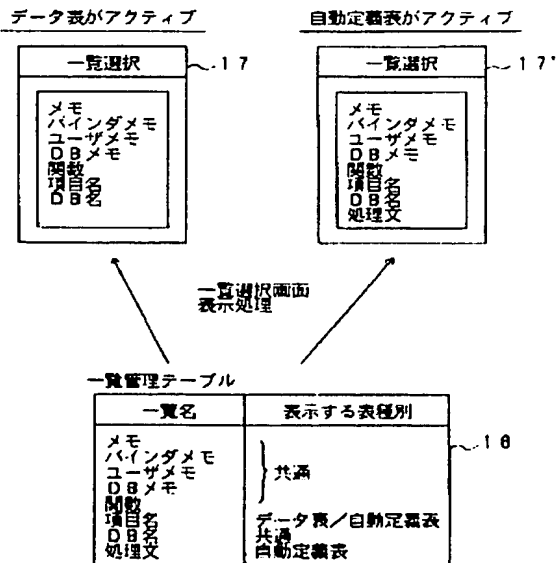
〔図1〕

本発明実施例による処理システムの基本構成図



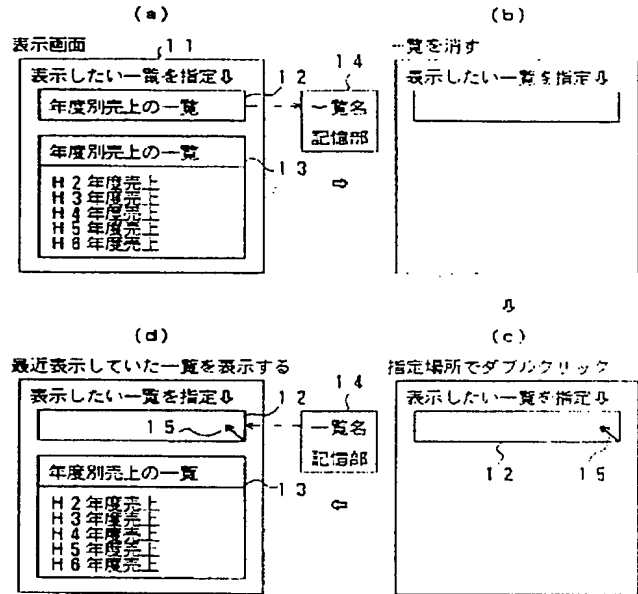
〔図6〕

本発明実施例による一覧選択画面の表示処理の説明図



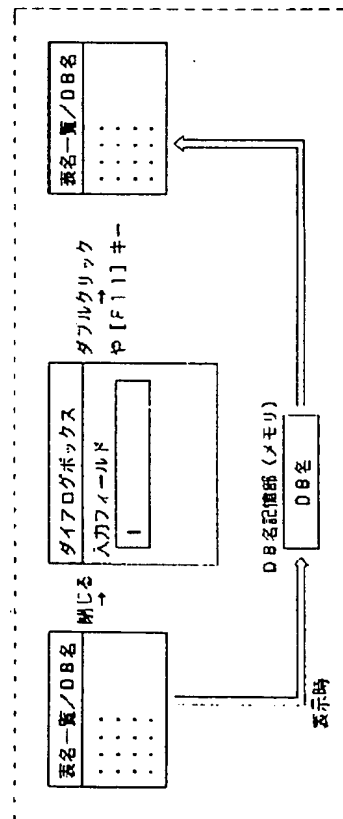
〔図2〕

本発明実施例による一覧画面の表示処理例の説明図



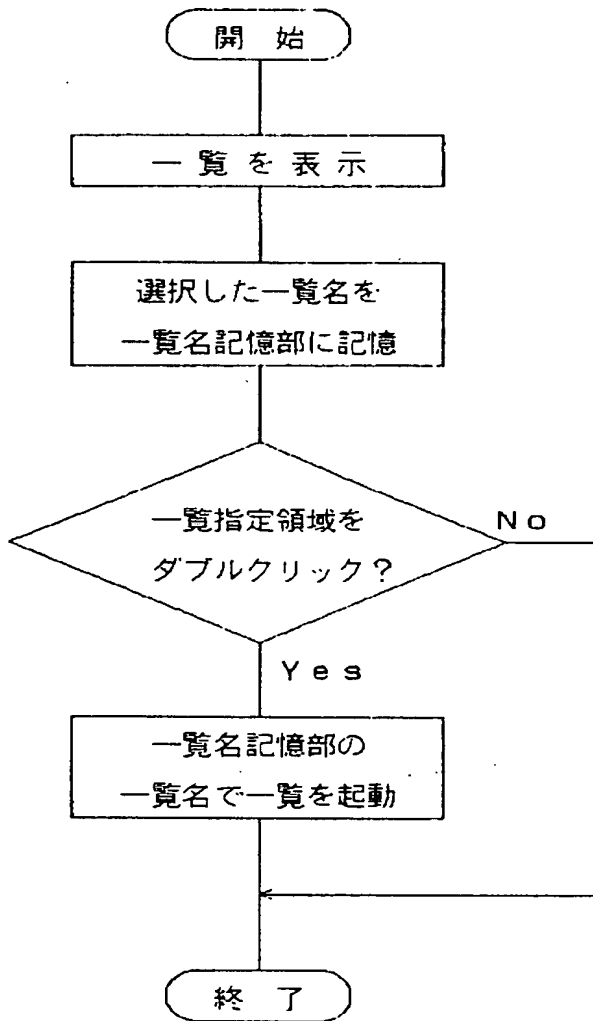
〔図4〕

表名一覧画面表示の具体例の説明図



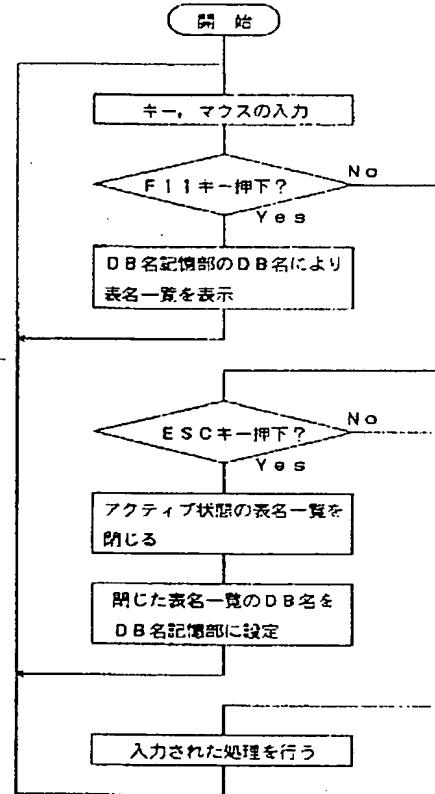
【図3】

一覧画面の表示処理のフロー



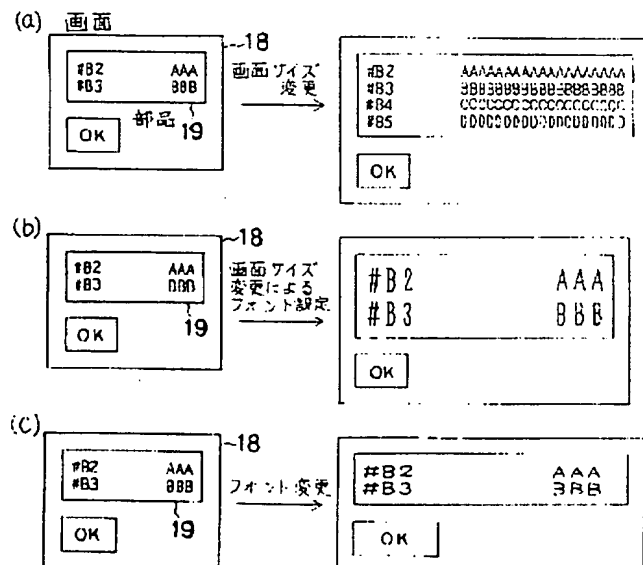
【図5】

専用キーによる一覧画面の表示制御のフロー



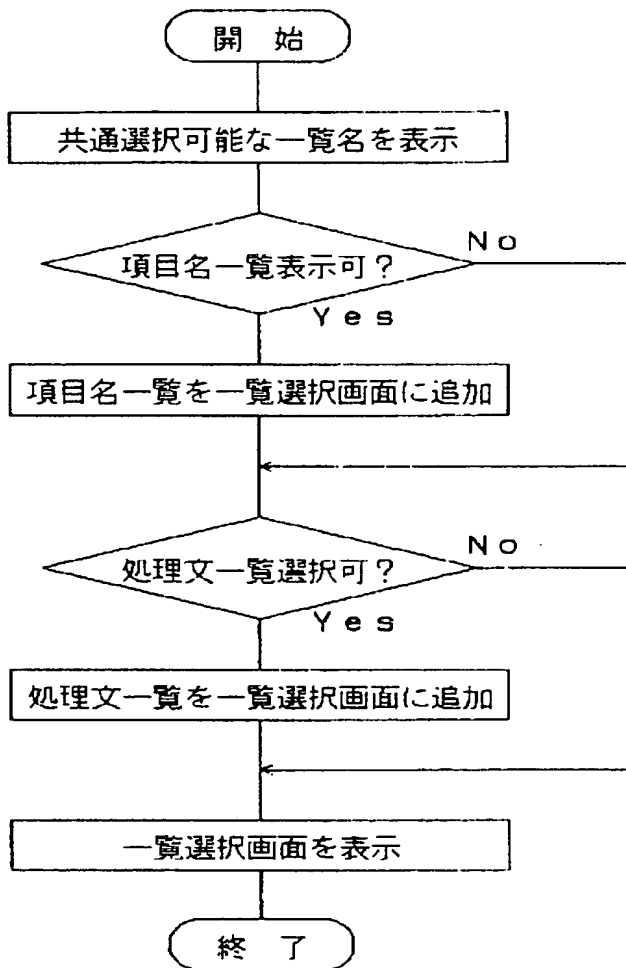
【図8】

本発明実施例による画面リサイズ、部品リサイズ、
フォントの変更を行う画面再構成処理の説明図



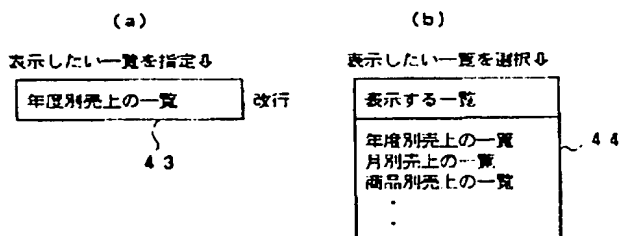
【図7】

一覧選択画面表示処理のフロー



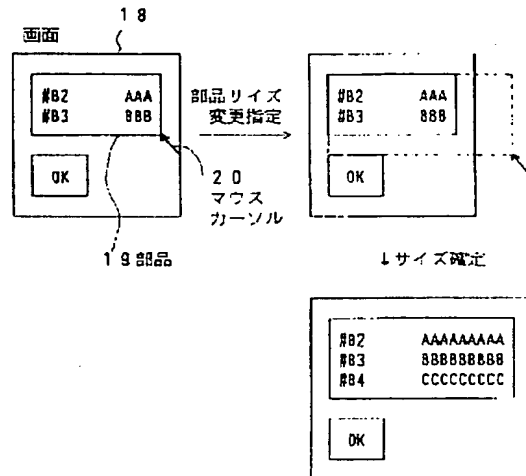
【図23】

従来のデータ一覧画面の表示指定方法の説明図



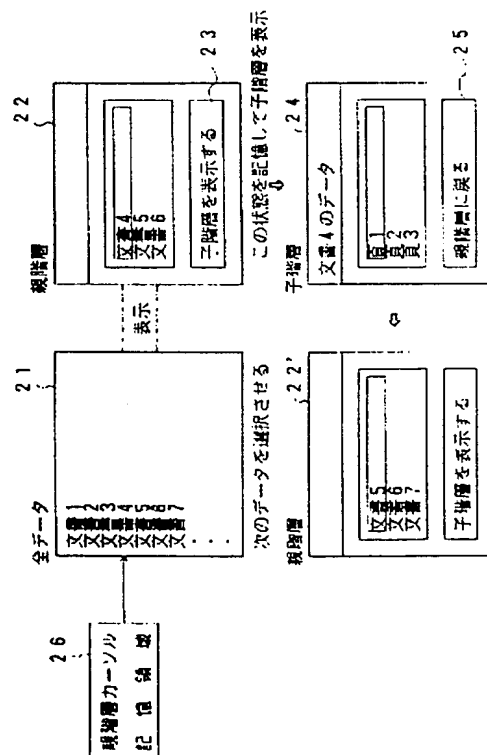
【図9】

部品サイズの変更指定による画面再構成例の説明図



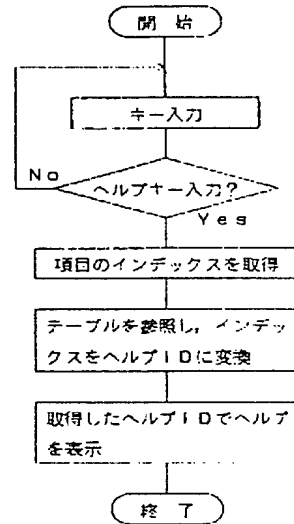
【図11】

本発明実施例による階層構造をもつ一覧データの表示処理の説明図

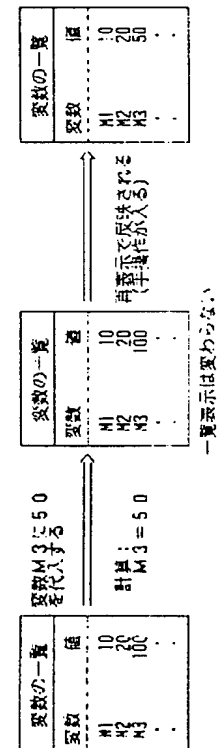


【例 18】

ヘルプ情報選択表示処理のフロー

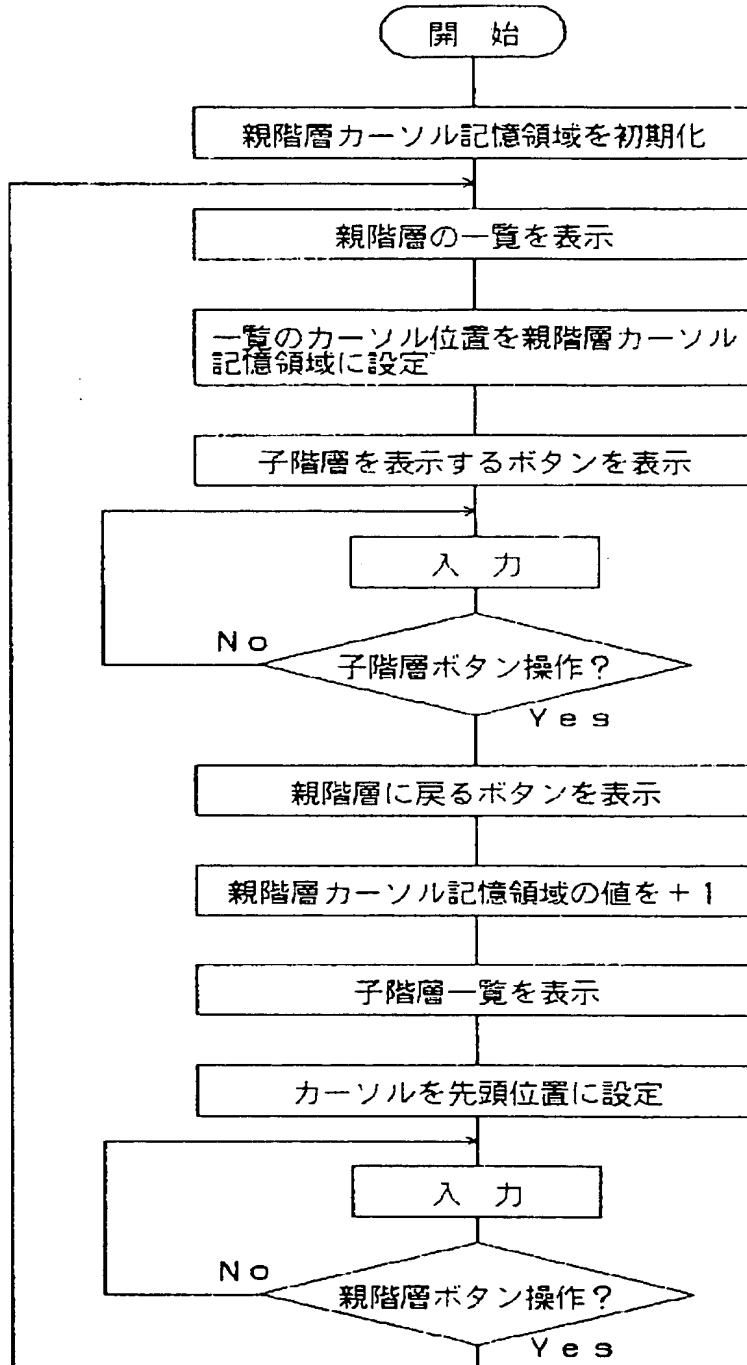


従来の変数一覧画面の表示処理の説明図



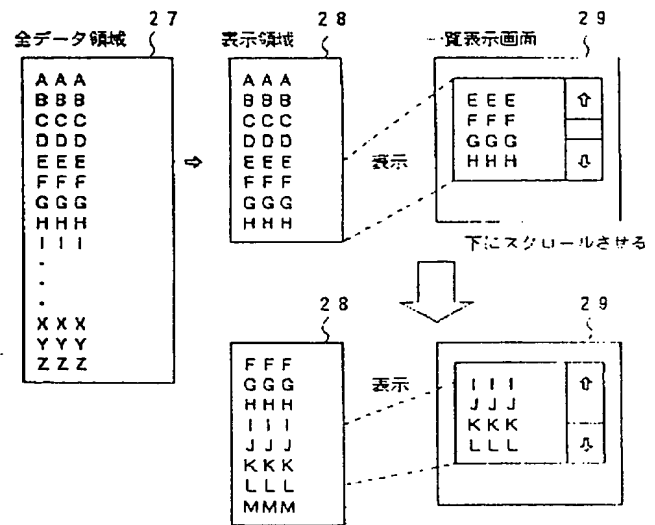
【図12】

階層構造をもつ一覧データの表示処理のフロー



【图 14】

大量一覧データの一覧画面のスクロール時の動作説明図



【图 17】

ヘルプ情報選択表示処理の説明図

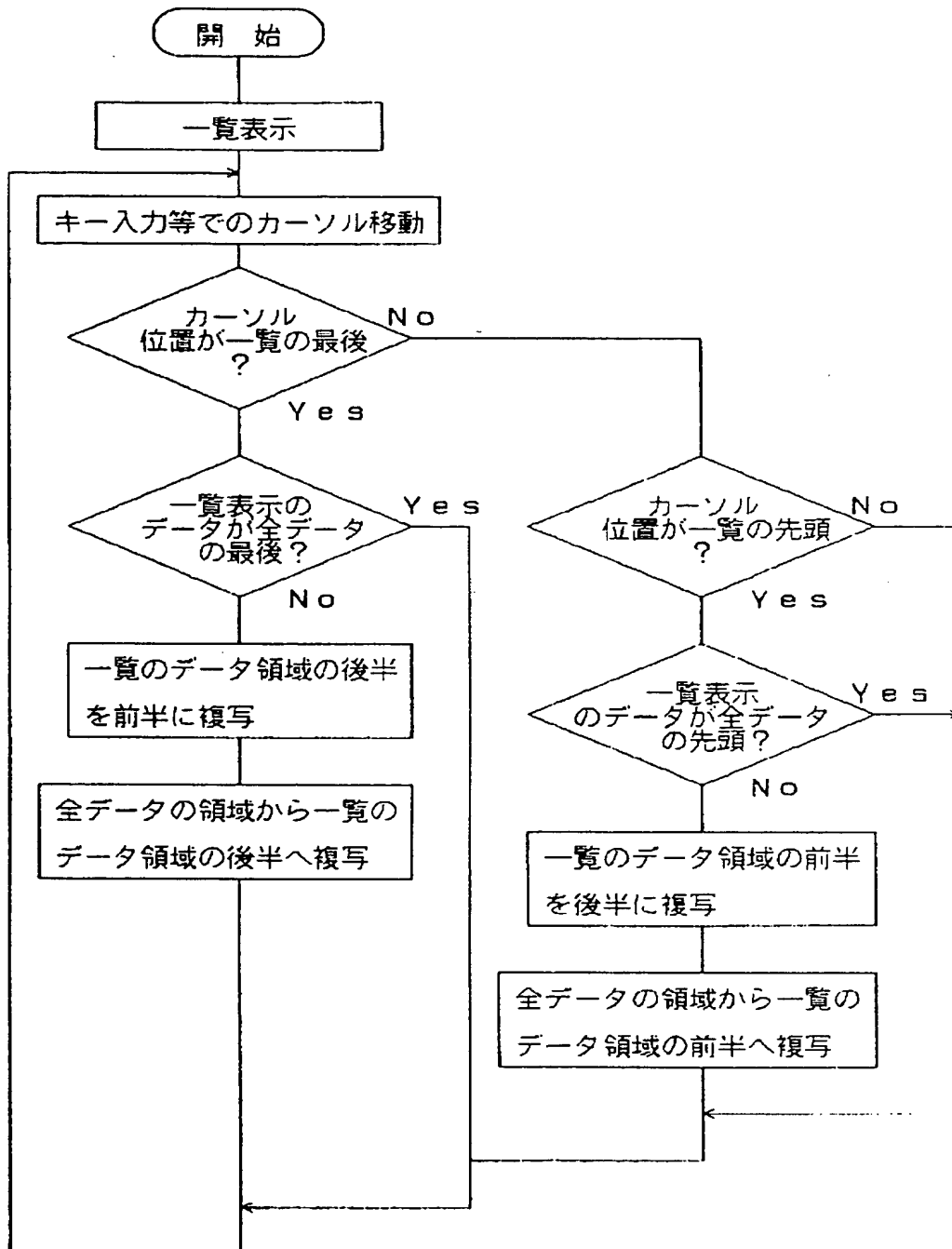
Figure 1 is a block diagram of the ABS control system. It shows the following components and their interconnections:

- 30 (ABS ECU):** The central control unit. It contains a microprocessor (MPU) and memory (ROM, RAM). It is connected to the Main Body (31), the ABS Solenoid Valve Driver (32), and the ABS Modulator (33).
- 31 (Main Body):** The vehicle chassis. It contains the ABS (Absolute Braking System) and the ABS Sensor. It is connected to the ABS ECU (30) and the ABS Modulator (33).
- 32 (ABS Solenoid Valve Driver):** A unit that drives the ABS solenoid valves. It is connected to the ABS ECU (30) and the ABS Modulator (33).
- 33 (ABS Modulator):** A unit that modulates the ABS pressure. It is connected to the ABS ECU (30), the ABS Solenoid Valve Driver (32), and the ABS Sensor (34).
- 34 (ABS Sensor):** A sensor that detects the wheel speed. It is connected to the ABS Modulator (33) and the Main Body (31).

The diagram also shows a 'ヘルプキー' (Help Key) connected to the ABS ECU (30) and a 'ヘルプ表示' (Help Display) connected to the Main Body (31).

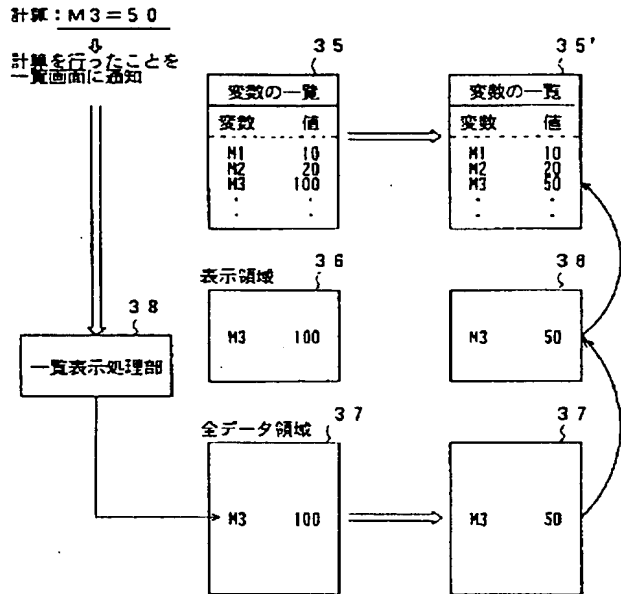
【図16】

大量一覧データの表示処理のフロー



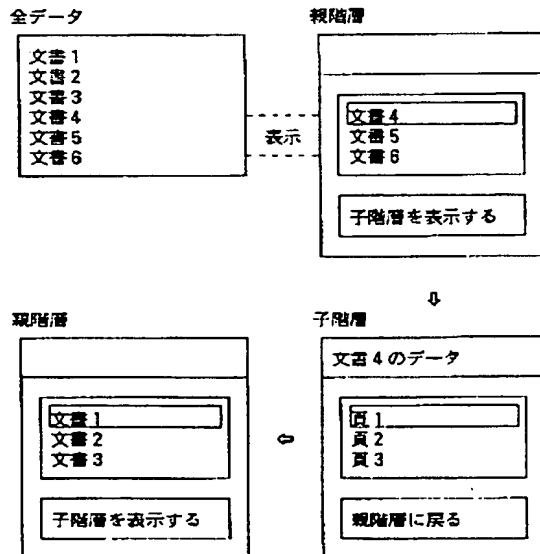
〔図19〕

本発明実施例による変数一覧画面の更新表示処理の説明図



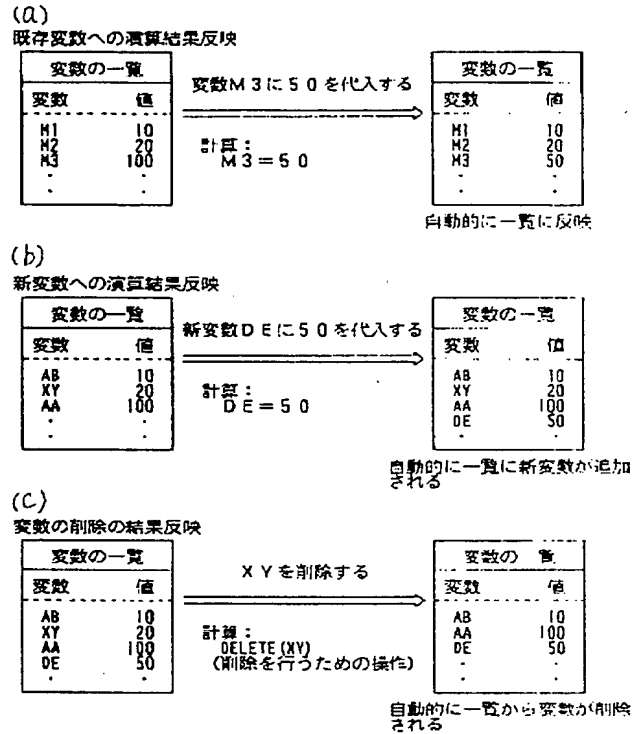
〔図24〕

従来の階層構造をもつ一覧画面の表示操作例の説明図



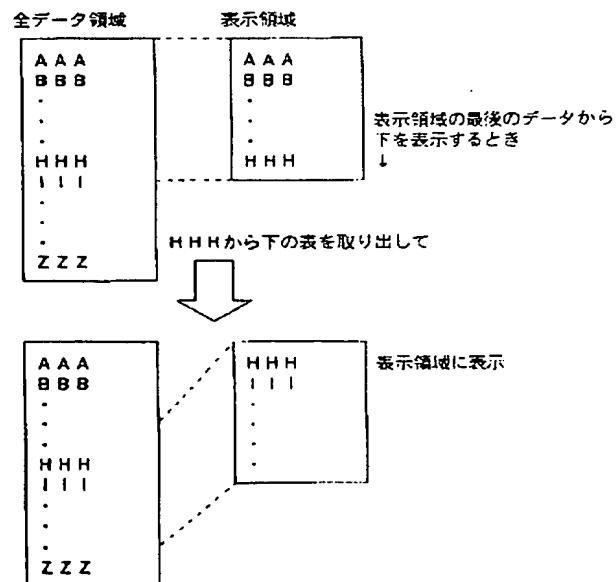
〔図20〕

変数画面の表示例の説明図



〔図25〕

従来の大量一覧データの表示処理の説明図

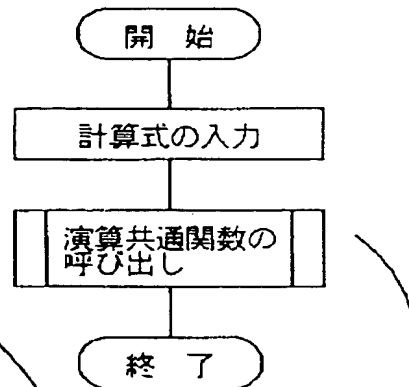
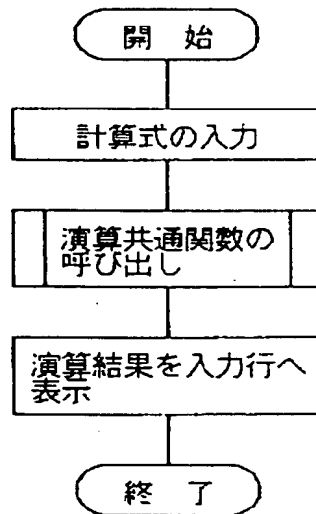


〔図21〕

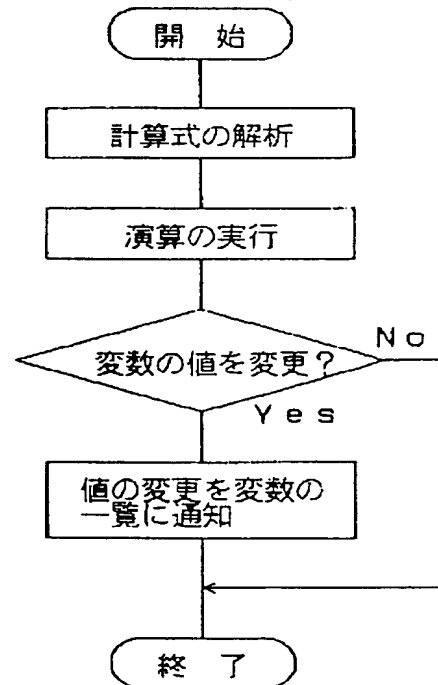
変数一覧画面の更新表示処理を起動する演算処理のフロー

(a) 計算コマンドによる実行

(b) 計算モードによる実行

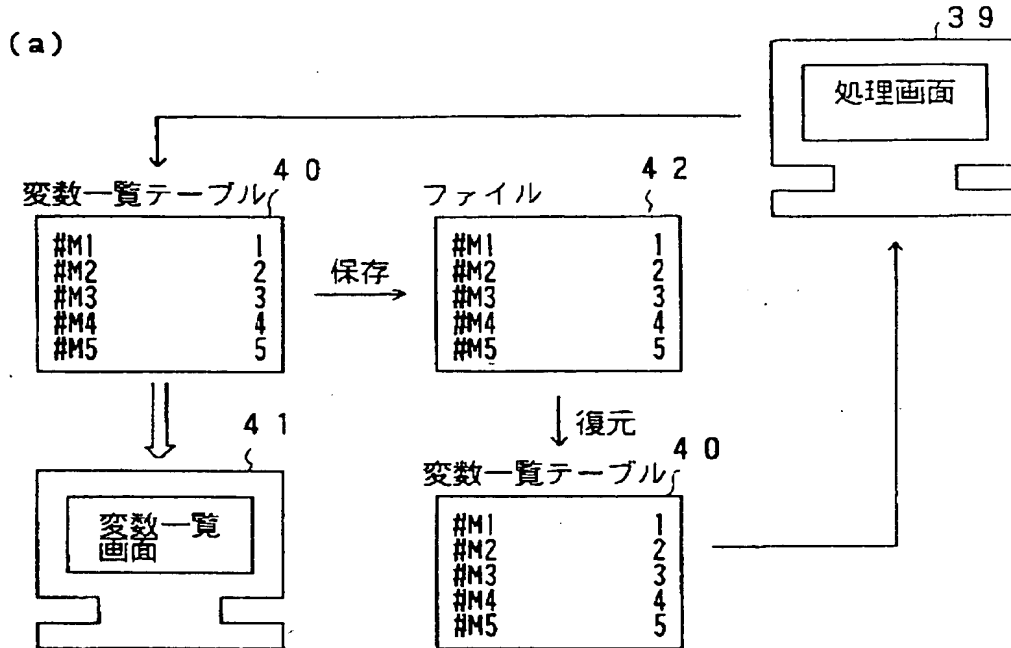


(c) 演算共通関数

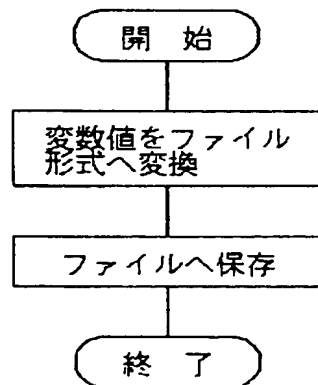


【図22】

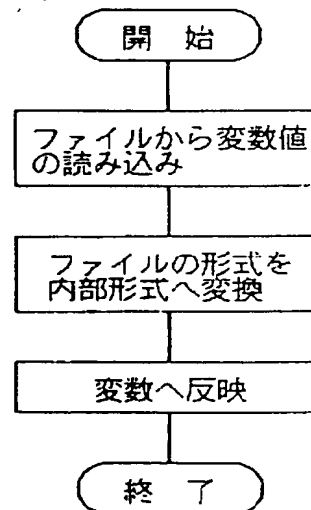
本発明実施例による変数一覧の変数の保存、復元処理の説明図



(b) 保存



(c) 復元



フロントページの続き

(72)発明者 酒井 裕偉
 大阪府大阪市北区堂島1丁目5番17号 ビ
 ーエフユー関西ソフトウェア株式会社内

(72)発明者 奥田 勝己
 大阪府大阪市北区堂島1丁目5番17号 ビ
 ーエフユー関西ソフトウェア株式会社内